Errors in long read assemblies can critically affect protein prediction

Mind the gaps – ignoring errors in long read assemblies can critically affect protein prediction

Amanda Warr[1] and Mick Watson[1]

[1]The Roslin Institute and Royal (Dick) School of Veterinary Studies, University of Edinburgh, Easter Bush, EH25 9RG

Long read, single molecule sequencing technologies are now routinely used for whole-genome sequencing and assembly. However, even after multiple rounds of correction, many errors can remain which can critically affect protein coding regions, resulting in significantly altered and often truncated protein predictions.

Second generation sequencing technologies have revolutionised biological research[1], largely driven by cheap, high-throughput short-read sequencing technologies[2]. However, it is now clear that these technologies often result in incomplete and highly fragmented assemblies[3] and may not be ideal for the assembly of large complex genomes. Technically and computationally, de novo genome assembly has been considered a "solved problem" for over a decade[4] – one simply needs reads that are longer than the longest repeat region, with sufficient depth and accuracy to detect overlaps between those reads.

Long-read, single-molecule technologies such as those produced by Pacific Biosciences[5] (PacBio) and Oxford Nanopore[6] (ONT) have the potential to sequence DNA molecules with lengths in the tens- or hundreds- of thousands of bases, enabling researchers to assemble large and complex repeats. However, both of these technologies have high per-read error rates (in the order of 5-15%), which has resulted in the development of "correction" algorithms. These attempt to use consensus base-calls, raw signal level data and/or shorter more accurate reads to correct long-read assemblies. Examples include Quiver and Arrow[7] for PacBio, Nanopolish[8] for ONT, and Pilon[9].

Published human genome assemblies using both PacBio[10,11] and Oxford Nanopore exist[12]. Pendleton *et al* reported a PacBio-only human genome assembly of NA12878 with a stated accuracy of 99.7%, whereas Koren *et al* report a polished PacBio human genome assembly of CHM1 with a stated accuracy of 99.8%. Most recently, in this journal Jain *et al* report the genome assembly of NA12878 using ONT's MinION device, a portable USB sequencer, generating the longest DNA reads ever sequenced. Accuracy after polishing is stated as 99.8%.

By most measures, these are highly accurate assemblies; however, in a genome of over three billion bases, each 0.1% of error represents over 3 million erroneous bases. Also, it is impossible to encode the accuracy of a genome assembly in a single percentage figure; in practice, large regions of any genome assembly are highly accurate, with most of the errors concentrated in repeat regions that suffer from a far lower accuracy than the stated figure[13–17]. The predominant errors in both PacBio and Nanopore sequencing technologies are insertions and deletions (indels)[18,19]. By introducing frameshifts and premature stop codons, these errors have the potential to critically affect translated regions, which rely on the fidelity of open-reading-frames to predict protein sequences from annotated transcripts.

Alternate assemblies of both NA12878[20] and CHM1[21] from short-read data are available, and can serve as a control for the above single molecule assemblies; where transcripts show evidence of indels in the long read assembly, but no evidence of indels in the short-read assembly of the same sample, we can be reasonably confident that those indels are errors specific to the long-read assembly.

Using all of these data, we sought to investigate the prevalence of insertion and deletion errors in the recently published Jain *et al* MinION/Illumina assembly of the human genome, including comparisons

to previously published long-read assemblies from PacBio data and short-read Illumina assemblies of the same cell lines.

Methods

The assemblies used in this analysis are given in table 1.

Accession	Technology	Chemistry	Coverage	Sample	Name used in this paper
GCA_900232925.1 Jain *et al*	Nanopore + Illumina polishing	R9.4 1D R9.4 1D ultra Illumina	27X 24X 55X	NA12878	NA12878.nano
GCA_001013985.1 Pendleton *et al*	PacBio + Quiver polishing	Pre P5-C3 P5-C3	24X 22X	NA12878	NA12878.pacb
GCA_000185165.1 Gnerre *et al*	Illumina paired-end + mate-pair + fosmid	Illumina	103X	NA12878	NA12878.ilum
chm1.round2.fasta* Koren *et al*	PacBio + Quiver polishing	P6-C4	142X	CHM1	CHM1.pacb
GCA_000306695.2 Steinberg *et al*	Illumina + BAC clone	Illumina	100X	CHM1	CHM1.ilum

* Available from http://gembox.cbcb.umd.edu/shared/canu/index.html

Table 1. A list of publicly available human genome assemblies used in this paper

We sought to minimise the computational burden of aligning all human transcripts to five human genome assemblies by introducing a filtering step. All exons containing protein-coding sequence were downloaded from Ensembl using BioMart and aligned to the above assemblies using BLAT. Short exons (<300bp) were removed, and alignments only considered where > 90% of the exon was contained within a single alignment. If an exon showed any evidence of insertions or deletions when compared against any of the assemblies, then the relevant transcript was added to a list of problematic transcripts. Any transcripts for which no BLAT alignment could be found were also added to this list, resulting in a total of 46423 "problematic" protein-coding transcripts for downstream analysis.

These were subsequently aligned to each assembly using splign[22], which attempts to find the best, full length spliced alignment between an RNA and its genomic sequence. Where splign suggested multiple potential hits, those producing protein coding alignments were prioritised, and then the best chosen as that with the lowest number of insertions/deletions, then the lowest number of mismatches. If a single best alignment could not be found, the transcript was rejected.

Alignments of the transcripts back against the GRCh38 reference genome were produced as control step. Of the 46423 transcripts, the following sets of transcripts were removed from downstream analysis: any transcript showing evidence of insertion/deletion errors in the GRCh38 splign analysis; any transcript ID annotated on an alternate haplotype of GRCh38; any transcript that did not map to the correct location in the GRCh38 splign analysis. Additionally, for NA12878 only, transcripts from the Y chromosome were removed (NA12878 originates from a female sample; CHM1 has been shown

to be male[23]). This resulted in 40949 transcripts for the NA12878 analyses, and 41035 for the CHM1 analyses.

For comparisons of long-read assemblies with their short-read counterparts, only transcripts with a near full-length (>80%) alignment in both assemblies were considered. Transcripts that show evidence of insertion/deletions in the single molecule assembly but not in the short-read assembly of the same sample were counted as errors.

Results

A summary of the results for each assembly can be seen in table 2, and a comparison of the single molecule assemblies with their short-read counterparts in table 3.

Assembly name	Input transcripts	# aligned	Full length	Near full length (>80%)	# total transcripts with indels	# total genes with indels
NA12878.nano	40949	34665	29440	34278	8478	3960
NA12878.pacb	40949	34606	29277	34146	25127	10736
NA12878.ilum	40949	31927	27131	31496	901	589
CHM1.pacb	41035	36128	30939	35744	1342	744
CHM1.ilum	41035	36487	31273	36104	587	397

Table 2 alignment statistics for all assemblies

Except for NA12878.ilum (which has fewer) all assemblies had similar numbers of total, full-length and near-full length mRNA alignments. Compared with their short-read counterparts, a naïve comparison shows that the Jain *et al* and Pendelton *et al* assemblies are massively enriched for indel errors (7x and 18x genes affected respectively). In contrast, the Koren *et al* assembly of CHM1 appears only slightly enriched for indel errors (1.9x genes affected).

Single-molecule assembly	Short-read control	# transcripts with indel errors	# genes with indel errors
NA12878.nano	NA12878.ilum	5929	2746
NA12878.pacb	NA12878.ilum	20816	8983
CHM1.pacb	CHM1.ilum	845	413

Table 3. Remaining indel errors in single molecule assemblies after removal of transcripts that show evidence of indels in the short-read assembly.

After subtraction of transcripts that show evidence of indel errors in the control short-read assemblies of the same sample, we are left with indel transcripts unique to the single molecule assemblies. The highest number of errors occurs in the Pacbio-only assembly of Pendelton *et al*, with 8983 protein coding genes predicted to be disrupted by insertions/deletions. Next is the polished nanopore assembly of NA12878 by Jain *et al*, with 2746 protein coding genes affected. Finally, the polished pacbio genome of Koren *et al* shows the best statistics; however, there are still 413 protein coding genes with indel errors in this assembly, broadly consistent with estimates of errors in other single-molecule assemblies of CHM1 reported in the literature[24].

Full results of the three comparisons reported in table 3 can be found in supplementary tables 1-3. Specific examples of alignments with indels are available in the supplementary information.

Discussion

Many factors influence genome assembly quality, including the underlying complexity of the genome in question, the ploidy of the cells being sequenced, the quality and accuracy of the technology being used to sequence the genome, the version and chemistry of that technology, the amount of sequence coverage generated, the length of the reads generated, the accuracy of tools used to assemble the genome and the accuracy of tools used to correct errors post-assembly, plus any manual steps used to correct errors the software tools cannot.

In this paper we assessed three long-read human genome assemblies for remaining insertion/deletion errors. All three assemblies reported accuracies between 99.7-99.8%, which may lead researchers to believe they are of a similar quality. Our analysis shows they are anything but.

Initial reports of the R7 MinION pore suggested 1st pass accuracies around 70-80%[18,25], and 2D (where each DNA strand is read twice and the consensus taken) accuracies around 85%[25,26]. The R7 pore is no longer available, nor is the 2D method; Jain *et al* report read accuracies in the region of 86% for a more recent pore (R9.4) and 1D sequence reads. MinION reads totally 51X coverage of the human genome were used to create the assembly, and 55X coverage Illumina reads used to polish remaining errors. Despite this, the assembly contains a significant number of indel errors, with 5929 transcripts and 2746 genes affected.

PacBio data has also undergone improvements, with raw read accuracies improving from 82% to 87% for later chemistries[27], which also tend to produce longer reads. Pendleton *et al* used a total of 46X coverage PacBio reads generated on the older P5-C3 chemistry to produce their assembly, and carried out one round of Quiver polishing. Unfortunately, in terms of indels, this has produced a very flawed assembly, with 20816 transcripts and 8983 protein-coding genes predicted to contain indel errors. Both the P5-C3 sequencing chemistry and Quiver have now been replaced, by P6-C4 and Arrow respectively.

There is a substantial improvement between the PacBio assembly produced by Koren *et al* compared to that produced by Pendleton *et al*. This is perhaps not surprising – as well as benefitting from longer and more accurate reads of the P6-C4 chemistry, the group generated 142X coverage and used two rounds of Quiver polishing. The assembly tool used, Canu, includes an at least one round of consensus-based read-correction. CHM1 is also a haploid cell line, which means the assembly and correction algorithms do not have to deal with the added complexity of differences between haplotypes[28]. Together, these improvements in data quality and bioinformatics explain the observed improvement. Without doubt the Koren *et al* assembly is highly accurate, yet there remain insertion/deletion errors affecting 845 protein coding transcripts and 413 protein coding genes.

Although the PacBio assembly produced by Pendleton *et al* is unlikely to be viewed as anything other than a "proof of concept", the large numbers of errors in that assembly serve as a warning to those trying to assemble genomes with lower quality data, lower coverage, and insufficient assembly and polishing work. The Koren *et al* assembly proves that it is possible to reduce the number of erroneous protein-coding regions to a few hundred, but it is important to note the resources and skills needed to do so.

The nanopore assembly by Jain *et al* benefitted from Pilon correction with short Illumina reads. However, many indels remain because of the problems inherent with mapping short Illumina reads to repetitive sequences (which includes gene families). If reads do not map, or map to multiple locations (a known issue in RNA-Seq[29]), then it can be more difficult to correct erroneous bases. Again, this assembly mainly exists as a "proof of concept", but many other research groups are undoubtedly

engaged in genome assembly using nanopore data, and the high number of indel errors in protein coding regions shown here (largely unaddressed in Jain *et al*) should serve as a warning to those groups to pay particular attention to insertion/deletion errors.

This analysis is not intended to be a comparison of sequencing technologies, nor should it be interpreted as such. Rather, it is an attempt to use published single molecule sequencing assemblies of the human genome to demonstrate that insertion/deletion errors remain prevalent, many of which can critically affect protein coding transcripts and genes. The human genome serves as a useful model for studying assembly accuracy given the availability of multiple public assemblies from the same samples (e.g. Genome in a Bottle24) and the availability of high quality annotation for the reference genome, GRCh38. The transcripts and genes identified in this study may be used as a focus for the improvement of assembly correction and improvement algorithms.

These results should not be considered a criticism of either PacBio or Oxford Nanopore, both of which are highly accurate technologies; nor should they be considered a criticism of Pendleton *et al*, Jain *et al* or Koren *et al*, all of which are ground-breaking pieces of research. Rather, the results indicate that even after multiple rounds of polishing, critical errors remain in single molecule assemblies that can critically affect protein predictions. This conclusion has ramifications across the biological and medical sciences, for those researchers seeking to sequence genomes (and seek funding to sequence genomes) using single molecule technologies. For those seeking to push long-read technologies into human clinical practice, the prevalence of indel errors remains a significant obstacle.

We are not suggesting that short-reads are a good alternative to long-reads when assembling a large or complex genome. Long reads have revolutionised genome assembly, and we believe they should be the starting point for all new genome assembly projects. Detailed assembly statistics for the five assemblies used in this paper can be found in Supplementary Table 4. NA12878.ilum, despite using 3 different types of long-range "jumping" libraries, has the shortest length, the largest number of gaps and the second lowest N50. Despite impressive statistics, CHM1.ilum is not typical of short-read assemblies as a reference-guided approach was used.

To obtain the best possible assembly, it is important to use high quality, high coverage sequencing data from one of the long-read technologies. Inclusion of data from multiple technologies can help improve assembly quality. It is important to incorporate multiple rounds of assembly polishing into downstream analyses, and to perform additional checks for remaining indels and errors. These additional checks should include alignment of known proteins and cDNA/mRNA sequences against the genome to check for genic indels, manual inspection of genomic alignments and, where necessary, manual fixing of errors that the correction algorithms miss. It is known that assembly quality has a huge impact on genome and gene annotation[30], and our work here provides further evidence that we must improve existing tools and build new tools that enable correction of genomes and undertake manual correction/curation where required.

A pipeline to reproduce the above analysis can be found at:

https://github.com/WatsonLab/sm_assemblies

References

1. Goodwin, S., Mcpherson, J. D. & Mccombie, W. R. Coming of age : ten years of next-generation sequencing technologies. *Nat. Publ. Gr.* **17,** 333–351 (2016).

2. Watson, M. Illuminating the future of DNA sequencing. *Genome Biol.* **15,** 108 (2014).

3. Bradnam, K. R. *et al.* Assemblathon 2: evaluating de novo methods of genome assembly in three vertebrate species. *Gigascience* **2,** 10 (2013).

4. Myers, E. W. The fragment assembly string graph. *Bioinformatics* **21,** ii79-ii85 (2005).

5. Eid, J. *et al.* Real-time DNA sequencing from single polymerase molecules. *Science* **323,** 133–8 (2009).

6. Loman, N. J. N. J. & Watson, M. Successful test launch for nanopore sequencing. *Nat. Methods* **12,** 303–4 (2015).

7. Chin, C.-S. *et al.* Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data. *Nat. Methods* **10,** 563–569 (2013).

8. Simpson, J. jts/nanopolish: Signal-level algorithms for MinION data. Available at: https://github.com/jts/nanopolish.

9. Walker, B. J. *et al.* Pilon: An Integrated Tool for Comprehensive Microbial Variant Detection and Genome Assembly Improvement. *PLoS One* **9,** e112963 (2014).

10. Koren, S. *et al.* Canu: scalable and accurate long-read assembly via adaptive *k* -mer weighting and repeat separation. *Genome Res.* **27,** 722–736 (2017).

11. Pendleton, M. *et al.* Assembly and diploid architecture of an individual human genome via single-molecule technologies. *Nat. Methods* **12,** 780–786 (2015).

12. Jain, M. *et al.* Nanopore sequencing and assembly of a human genome with ultra-long reads. *Nat. Biotechnol.* (2018). doi:10.1038/nbt.4060

13. Treangen, T. J. & Salzberg, S. L. Repetitive DNA and next-generation sequencing: computational challenges and solutions. *Nat. Rev. Genet.* **13,** 36–46 (2011).

14. Pop, M. & Salzberg, S. L. Bioinformatics challenges of new sequencing technology. *Trends Genet.* **24,** 142–149 (2008).

15. Phillippy, A. M., Schatz, M. C. & Pop, M. Genome assembly forensics: finding the elusive mis-assembly. *Genome Biol.* **9,** R55 (2008).

16. Gajer, P., Schatz, M. & Salzberg, S. L. Automated correction of genome sequence errors. *Nucleic Acids Res.* **32,** 562–9 (2004).

17. Salzberg, S. L. & Yorke, J. A. Beware of mis-assembled genomes. *Bioinformatics* **21,** 4320–4321 (2005).

18. Jain, M. *et al.* Improved data analysis for the MinION nanopore sequencer. *Nat. Methods* **12,** 351–6 (2015).

19. Weirather, J. L. *et al.* Comprehensive comparison of Pacific Biosciences and Oxford Nanopore Technologies and their applications to transcriptome analysis. *F1000Research* **6,** 100 (2017).

20. Gnerre, S. *et al.* High-quality draft assemblies of mammalian genomes from massively parallel sequence data. *Proc. Natl. Acad. Sci.* **108,** 1513–1518 (2011).

21. Steinberg, K. M. *et al.* Single haplotype assembly of the human genome from a hydatidiform mole. *Genome Res.* **24,** 2066–76 (2014).

22. Kapustin, Y., Souvorov, A., Tatusova, T. & Lipman, D. Splign: algorithms for computing spliced alignments with identification of paralogs. *Biol. Direct* **3,** 20 (2008).

23. Fisher, R. A., Khatoon, R., Paradinas, F. J., Roberts, A. P. & Newlands, E. S. Repetitive complete

hydatidiform mole can be biparental in origin and either male or female. *Hum. Reprod.* **15,** 594–598 (2000).

24. Schneider, V. A. *et al.* Evaluation of GRCh38 and de novo haploid genome assemblies demonstrates the enduring quality of the reference assembly. *Genome Res.* **27,** 849–864 (2017).

25. Quick, J., Quinlan, A. R. & Loman, N. J. A reference bacterial genome dataset generated on the MinION™ portable single-molecule nanopore sequencer. *Gigascience* **3,** 22 (2014).

26. Risse, J. *et al.* A single chromosome assembly of Bacteroides fragilis strain BE1 from Illumina and MinION nanopore sequencing data. *Gigascience* **4,** 60 (2015).

27. Koren, S. & Phillippy, A. M. One chromosome, one contig: complete microbial genomes from long-read sequencing and assembly. *Curr. Opin. Microbiol.* **23,** 110–120 (2015).

28. Koren, S. *et al.* Complete assembly of parental haplotypes with trio binning. *bioRxiv* 271486 (2018). doi:10.1101/271486

29. Robert, C. & Watson, M. Errors in RNA-Seq quantification affect genes of relevance to human disease. *Genome Biol.* **16,** 177 (2015).

30. Florea, L., Souvorov, A., Kalbfleisch, T. S. & Salzberg, S. L. Genome Assembly Has a Major Impact on Gene Content: A Comparison of Annotation in Two Bos Taurus Assemblies. *PLoS One* **6,** e21400 (2011).

Supplementary Table Descriptions

Supplementary table 1

Summary of splign alignments comparing NA12818.nano with NA12878.ilum. Columns: vtid = versioned ensemble transcript id; tid = ensemble transcript id; gid = ensemble gene id; len = length of the transcript; nan.tid = the query ID from splign; nan.hit = the hit ID from splign; nan.start = the start of the alignment in transcript co-ordinates; nan.len = the length of the alignment; nan_over_len = ratio of alignment length to query length; nan.mis = the number of mismatch events; nan.misb = the number of bases in mismatch events; nan.ins = the number of insertion events; nan.insb = the number of bases in insertion events; nan.del = the number of deletion events; nan.delb = the number of bases in deletion events; nan.indels = the total number of indel events; nan.sequence = the predicted protein sequence from the nanopore alignment; ilm.tid = the query ID from splign; ilm.hit = the hit ID from splign; ilm.start = the start of the alignment in transcript co-ordinates; ilm.len = the length of the alignment; ilm_over_len = ratio of alignment length to query length; ilm.mis = the number of mismatch events; ilm.misb = the number of bases in mismatch events; ilm.ins = the number of insertion events; ilm.insb = the number of bases in insertion events; ilm.del = the number of deletion events; ilm.delb = the number of bases in deletion events; ilm.indels = the total number of indel events; ilm.sequence = the predicted protein sequence from the illumina alignment

Supplementary table 2

Summary of splign alignments comparing NA12878.pacb with NA12878.ilum. Columns: vtid = versioned ensemble transcript id; tid = ensemble transcript id; gid = ensemble gene id; len = length of

the transcript; pacb.tid = the query ID from splign; pacb.hit = the hit ID from splign; pacb.start = the start of the alignment in transcript co-ordinates; pacb.len = the length of the alignment; pacb_over_len = ratio of alignment length to query length; pacb.mis = the number of mismatch events; pacb.misb = the number of bases in mismatch events; pacb.ins = the number of insertion events; pacb.insb = the number of bases in insertion events; pacb.del = the number of deletion events; pacb.delb = the number of bases in deletion events; pacb.indels = the total number of indel events; pacb.sequence = the predicted protein sequence from the pacbio alignment; ilm.tid = the query ID from splign; ilm.hit = the hit ID from splign; ilm.start = the start of the alignment in transcript co-ordinates; ilm.len = the length of the alignment; ilm_over_len = ratio of alignment length to query length; ilm.mis = the number of mismatch events; ilm.misb = the number of bases in mismatch events; ilm.ins = the number of insertion events; ilm.insb = the number of bases in insertion events; ilm.del = the number of deletion events; ilm.delb = the number of bases in deletion events; ilm.indels = the total number of indel events; ilm.sequence = the predicted protein sequence from the illumina alignment

Supplementary table 3

Summary of splign alignments comparing CHM1.pacb with CHM1.ilum. Columns: vtid = versioned ensemble transcript id; tid = ensemble transcript id; gid = ensemble gene id; len = length of the transcript; pacb.tid = the query ID from splign; pacb.hit = the hit ID from splign; pacb.start = the start of the alignment in transcript co-ordinates; pacb.len = the length of the alignment; pacb_over_len = ratio of alignment length to query length; pacb.mis = the number of mismatch events; pacb.misb = the number of bases in mismatch events; pacb.ins = the number of insertion events; pacb.insb = the number of bases in insertion events; pacb.del = the number of deletion events; pacb.delb = the number of bases in deletion events; pacb.indels = the total number of indel events; pacb.sequence = the predicted protein sequence from the pacbio alignment; ilm.tid = the query ID from splign; ilm.hit = the hit ID from splign; ilm.start = the start of the alignment in transcript co-ordinates; ilm.len = the length of the alignment; ilm_over_len = ratio of alignment length to query length; ilm.mis = the number of mismatch events; ilm.misb = the number of bases in mismatch events; ilm.ins = the number of insertion events; ilm.insb = the number of bases in insertion events; ilm.del = the number of deletion events; ilm.delb = the number of bases in deletion events; ilm.indels = the total number of indel events; ilm.sequence = the predicted protein sequence from the illumina alignment

Supplementary table 4

Summary statistics for the 5 assemblies calculated using assembly-stats. “Length (Gb)” = length of the assembly in gigabases; “# seqs” = number of unique sequences in the assembly; “Gaps” = number of gaps; “N” = total number of N bases; “N50 (Mb)” the assembly N50 in megabases